NUMERICAL LINEAR ALGEBRA WITH APPLICATIONS
Numer. Linear Algebra Appl. 2000; **00**:1–6 *Prepared using* nlaauth.cls *[Version: 2002/09/18 v1.02]*

A cache-oblivious self-adaptive full multigrid method

M. Mehl[1,*], T. Weinzierl[1], Chr. Zenger[1]

[1] *Institut für Informatik, TU München, Boltzmannstraße 3, 85748 Garching, Germany*

SUMMARY

This paper presents a new efficient way to implement multigrid algorithms on adaptively refined grids. To cope with todays demands in high performance computing, we cannot do without such highly sophisticated numerical methods. But if we do not implement them very carefully, we lose a lot of efficiency in terms of memory usage: using trees for the storage of hierarchical multilevel data causes a large amount of nonlocal (in terms of the physical memory space) data accesses, and often requires the storage of pointers to neighbors to allow the evaluation of discrete operators (difference stencils, restrictions, interpolations, etc.). The importance of this problem becomes clear if we remember that storage and not the CPUs is the bottleneck on modern computers.

We established a cache-oblivious and storage-minimizing algorithm based on the concept of space-tree grids combined with a cell-oriented operator evaluation, a linear ordering of grid cells along a space-filling curve, and a sophisticated construction of linearly processed data structures for vertex data. In this context, we could show that the implementation of a dynamically adaptive F-cycle is, first, very natural and, second, does not cause any overhead in terms of storage usage and access as adaptivity and multilevel data do not disturb the linear processing order of our data structures.

KEY WORDS: cache-efficiency, multigrid, dynamical adaptivity, space-tree, space-filling curve

1. INTRODUCTION

In general, the problem of finding efficient implementations for numerically efficient methods is well known and there are numerous solution strategies

and the auxiliary variable

$$r^2 := \left(x_1 - \frac{1}{3}\right)^2 + \left(x_2 - \frac{1}{3}\right)^2 + \left(x_3 - \frac{1}{3}\right)^2,$$

which has the analytical solution (see also Figure 6 for the two-dimensional analogon)

$$u = \frac{1}{\sinh(128\pi)} \sinh(64\pi(2 - r^2)) \text{ on } \partial]0;[1^3$$

for an appropriate definition of boundary values.

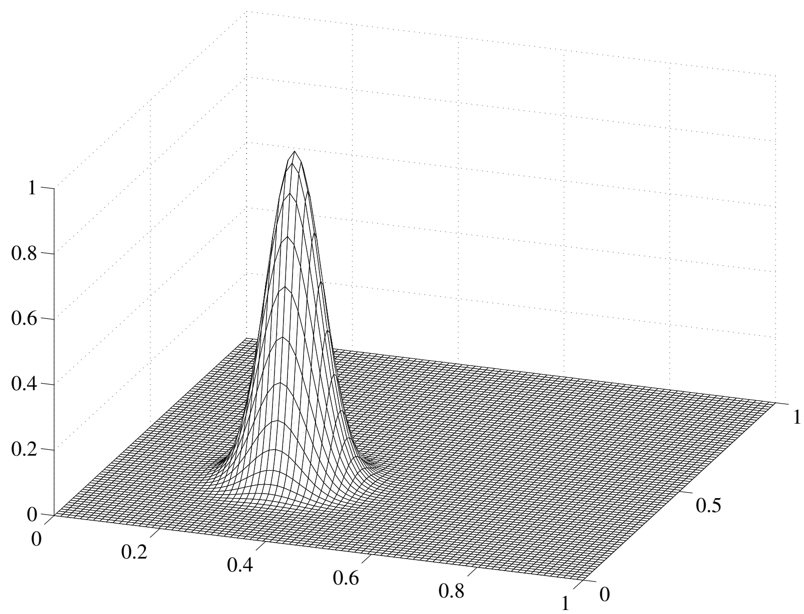

Figure 6. Graph of the two-dimensional function $\frac{1}{\sinh(128\pi)} \sinh\left(64\pi\left(2 - \left(x_1 - \frac{1}{3}\right)^2 - \left(x_2 - \frac{1}{3}\right)^2\right)\right)$ on the unit square $[0;1]^2$ (taken from [23]).

5.3.1. Cache-performance For all examples computed and for all three adaptivity criteria, the level 2 cache hit rates, that is the relation between the number of successfull data accesses on the level two cache and the total number of all data accesses was above 99.8%.

5.3.2. Relation Accuracy – # Degrees of Freedom The numerical efficieny of adaptivity criteria can be measured by the dependence between the number of degrees of freedom and the achieved accuracy. The generated grids strongly depend on the used criterion (see Figure 7).

Figure 8 shows the development of errors in dependence on the adaptivity criterion. Although the linear surplus generates locally deeper refined grids (Figure 7), even the local accuracy at

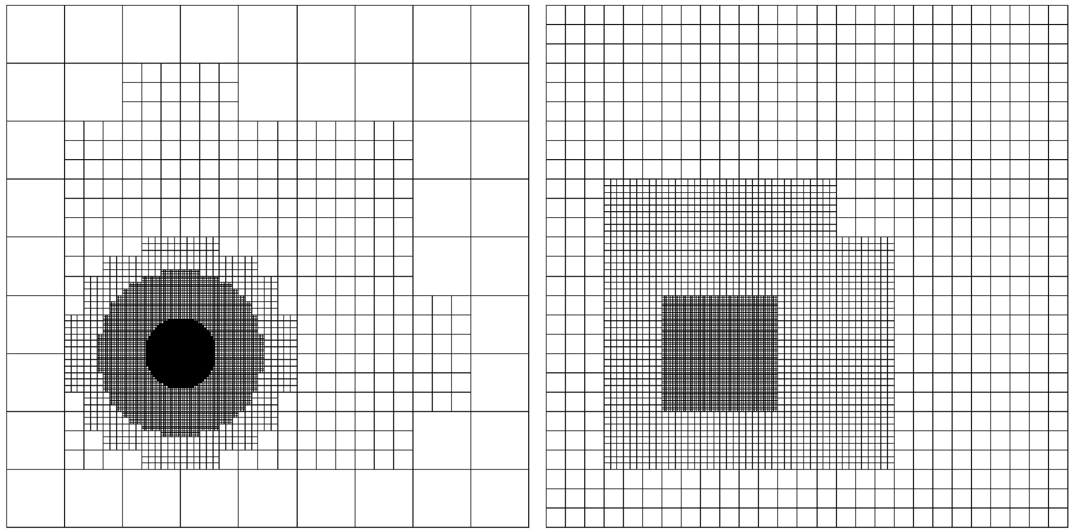

Figure 7. Two-dimensional projection of the grids resulting from the application of the linear surplus (left) and the τ-criterion (right) (taken from [23]).

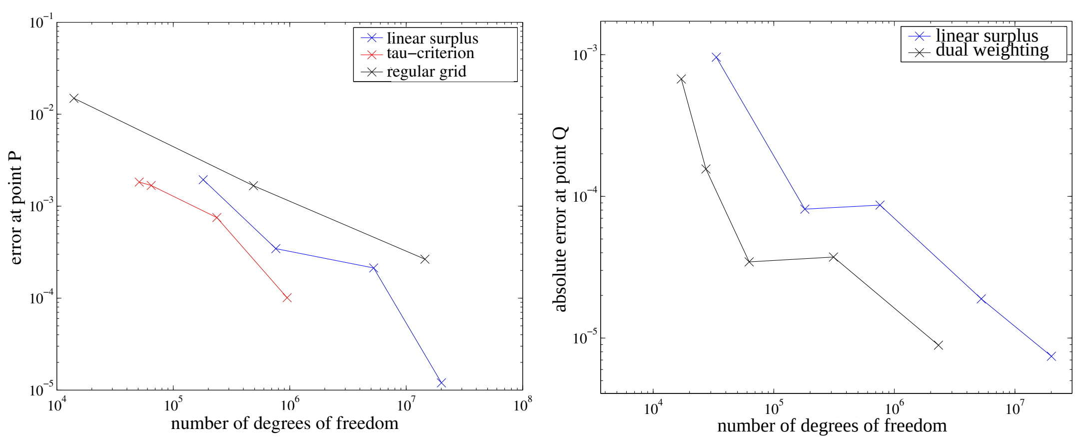

Figure 8. Absolute error at the point $P = \left(\frac{10}{27}, \frac{10}{27}, \frac{10}{27}\right)$ for regular grids and adaptive grids using the linear surplus or the τ-criterion (left); absolute error at the point $Q = \left(\frac{2}{3}, \frac{2}{3}, \frac{2}{3}\right)$ for adaptive grids using the linear surplus or the dually weighted linear surplus (right) (taken from [23]).

the tip of the peak of the analytical solution of the example equation is better for the τ-criterion. This shows the influence of the accuracy in the further surrounding on the accuracy at a certain point. Therefore, also the application of the dually weighted linear surplus brings a gain in terms of the relation accuracy versus needed number of degrees of freedom.

5.3.3. Runtime Considering the runtime per iteration and degree of freedom, we observed almost no differences between the linear surplus and the τ-criterion above a certain number of

	cache-efficiency	accuray	runtime	storage
linear surplus	$\geq 99.8\%$	depends	+	+
τ-criterion	$\geq 99.8\%$	depends	+	+
dual weighting	$\geq 99.8\%$	depends	-	-

Table VII. Summary of the comparison of the three implemented adaptivity criteria.

degrees of freedom. Only the dually weighted linear surplus caused an about $10 - 15\%$ longer runtime due to the extra-computation of the dual solution and its linear surplus.

A summary of the results of the comparison of our adaptivity criteria can be found in Table VII. The cache-efficiency is equally high for all of them, the judgement of the accuracy depends on the given example and on the objective function in terms of the error. The superiority of the linear surplus and the τ-crierion with respect to runtime and storage requirements is naturally given by the computational requirements of the criterion. Anyhow, the dual weighting is a very usefull tool to minimize general error functionals.

5.4. Behaviour for Singularities

To show the high flexibility of our approach, we solved a singular problem on the unit cube with a cubic cut-off (the three-dimensional analogon of an L-shape domain. Figure 9 shows the computational domain.

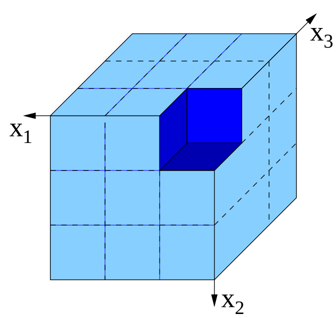

Figure 9. Unit cube with a cubic cut-off.

On this domain Ω, we solve the Poisson equation

$$\begin{aligned} -\Delta u &= 3\pi^2 \cdot \prod_{j=1}^{3} \sin(\pi x_j) \text{ in } \Omega \\ u &= 0 \text{ on } \partial\Omega. \end{aligned}$$

Table VIII shows the tremendously lower number of degrees of freedom that is needed with an adaptively refined grid in comparison to a regular grid to achieve the same accuracy (measured by the maximal discretization error τ_{max}).

To summarize the numerical results, we can say that we solve systems with up to 10^{10} degrees of freedom [20] on highly and dynamically adaptive grids with a computing time of less than $2 \cdot 10^{-6}$ seconds per iteration and degree of freedom (all measured on an AMD Athlon

Prepared using nlaauth.cls

	regular grid	adaptive grid
# degrees of freedom	$509,656$	$61,267$

Table VIII. Number of degrees of freedom needed for a regular grid and an (on the finest level) according to the τ-criterion adaptively refined grid to achieve a maximal discretization error $\tau_{max} = 1.1734 \cdot 10^{-3}$

XP 2400+ (1.9 GHz) processor with 256 KB cache and 1 GB RAM using the *gcc3.4* compiler with options `-O3 -Xw`.

6. CONCLUSION

As we could show in the previous sections, we established an algorithmic concept for the implementation of multigrid methods on dynamically adaptive grids, which fulfilled at the same time the numerical demands on a modern simulation code (multigrid performance, flexible adaptivity) and shows an efficient usage of hardware resources, in particular memory hierarchies. The latter is valid independently from the actual hardware parameters like for example chache size, cache-line length, associativity of the cache, etc. [19].

REFERENCES

1. Goedecker S, Hoisie A. *Performance Optimization of Numerically Intensive Codes*. SIAM, 2001.
2. Douglas C C. Caching in With Multigrid Algorithms: Problems in Two Dimensions. *Parallel Algorithms and Applications* 1996; **9**: 195-204.
3. Douglas C C, Hu J, Kowarschik M, Rüde U, Weiß C. Cache Optimization for Structured and Unstructured Grid Multigrid. *Electronic Transactions on Numerical Analysis* 2000; **10**: 21-40.
4. Kowarschik M. *Data Locality Optimizations for Iterative Numerical Agorithms and Cellular Automata on Hierachical Memory Architectures*. PhD thesis, Institut für Informatik, Universität Erlangen-Nürnberg, SCS Publishung House, 2004.
5. Kowarschik M, Weiß C. An Overview of Cache Optimization Techniques and Cache-Aware Numerical Algorithms. In: Meyer, Sanders, and Sibeyn (eds.), *Algorithms for Memory Hierarchies – Advanced Lectures*; Lecture Notes in Computer Science, **2625**: 213-232, Springer, 2003.
6. Weiß C. *Data Locality Optimizations for Multigrid Methods on Structured Grids*. Phd thesis, Institut für Informatik, TU München, 2001.
7. Sagan H. *Space-Filling Curves*. Springer: New York, 1994.
8. Prokop H. *Cache-Oblivious Algorithms*. Master thesis, Massachusetts Institute of Technology, 1999.
9. Frigo M, Leierson C E, Prokop H, Ramchandran S, Cache-oblivious algorithms. In: *Proceedings of the 40th Annual Sympoisium on Foundations of Computer Science*: 285-297, New York, 1999.
10. Demaine E D. Cache-Oblivious Algorithms and Data Structures. In: *Lecture Notes from the EEF Summer School on Massive Data Sets, University of Aarhus, Denmark, June 27-July 1*; Lecture Notes in Computer Science, Springer, 2002.
11. Griebel M, Zumbusch G W. Parallel multigrid in an adaptive PDE solver based on hashing and space-filling curves. *Parallel Computing* 1999; **25**: 827-843.
12. Griebel M, Zumbusch G. Hash based adaptive parallel multilevel methods with space-filling curves. In: Rollnik and Wolf (eds.), *NIC Series* 2002; **9**, 479-492.
13. Oden J T, Patra A, Feng Y. Domain decomposition for adaptive *hp* finite element methods. In: Keyes and Xu (eds.), *Domain decomposition methods in scientific and engineering computing; Proceedings of the 7th int. conf. on domain decomposition*; Contemporary Mathematics 1994; **180**: 203-214.
14. Patra A K, Long J, Laszloff A. Efficient Parallel Adaptive Finite Element Methods Using Self-Scheduling Data and Computations. In: Banerjee, Prasanna, and Sinha (eds.), *High Performance Computing – HiPC'99, 6th International Conference, Calcutta, India, December 17-20, 1999, Proceedings*; HiPC, Lecture Notes in Compter Science 1999; **1745**: 359-363.

15. Roberts S, Klyanasundaram S, Cardew-Hall M, Clarke W. A key based parallel adaptive refinement technique for finite element methods. In: Noye, Teubner, and Gill (eds.), *Proceedings Computational Techniques and Applications: CTAC '97*: 577-584, World Scientific:Singapore, 1998.
16. Zumbusch G W. *Adaptive Parallel Multilevel Methods for Partial Differential Equations.* Habilitationsschrift, Universität Bonn, 2001.
17. Aftosmis M J, Berger M J, Adomavivius G. *A Parallel Multilevel Method for Adaptively Refined Cartesian Grids with Embedded Boundaries*, American Institute of Aeronautics and Astronautics-2000-808, Aerospace Science Meeting and Exhibit, 38th, Reno, Nevada, Jan 10-13, 2000.
18. Braess D. *Finite Elements. Theory, Fast Solvers and Applications in Solid Mechanics.* Cambridge University Press, 2001.
19. Günther F. *Eine cache-optimale Implementierung der Finite-Elemente-Methode.* Doctoral thesis, Institut für Informatik, TU München, 2004.
20. Pögl M. *Entwicklung eines cache-optimalen 3D Finite-Element-Verfahrens für große Probleme.* Doctoral thesis, Institut für Informatik, TU München, 2004.
21. Hartmann J. *Entwicklung eines cache-optimalen Finite-Element-Verfahrens zur Lösung d-dimensionaler Probleme.* Diploma thesis, Institut für Informatik, TU München, 2005.
22. Günther F, Mehl M, Pögl M, Zenger Ch. A cache-aware algorithm for PDEs on hierarchical data structures based on space-filling curves. *SIAM Journal on Scientific Computing*, in review.
23. Dieminger N. *Kriterien für die Selbstadaption cache-effizienter Mehrgitteralgorithmen* . Diploma thesis, Institut für Informatik, TU München, 2005.
24. Fulton S R. On the accuracy of multigrid truncation error estimates. Electronic Transactions on Numerical Analysis 2003; **15**: 29-37.
25. Schneider S. *Adaptive Solution of Elliptic Partial Differential Equations by Hierarchical Tensor Product Finite Elements.* Doctoral thesis, Institut für Informatik, TU München, 2000.
26. Krahnke A. *Adaptive Verfahren höherer Ordnung auf cache-optimalen Datenstrukturen für dreidimensionale Probleme.* Doctoral thesis, Institut für Informatik, TU München, 2005.
27. Kowarschik M, Wei C. DiMEPACK - A Cache-Optimized Multigrid Library. In: Arabnia (ed.), *Proceedings of International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2001), Las Vegas, Nevada, USA* 2001; **I**.
28. Seward J, Nethercote N, Fitzhardinge J. cachegrind: a cache-miss profiler. `http://valgrind.kde.org/docs.html`

6.1. Copyright